Using Collaborative Engineering to Inform Collaboration Engineering

Lynne P. Cooper
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive Pasadena, CA 91109
lynne.p.cooper@jpl.nasa.gov

Abstract

Collaboration is a critical competency for modern organizations as they struggle to compete in an increasingly complex, global environment. A large body of research on collaboration in the workplace focuses both on teams, investigating how groups use teamwork to perform their task work, and on the use of information systems to support team processes ("collaboration engineering"). This research essay presents collaboration from an engineering perspective ("collaborative engineering"). It uses examples from professional and student engineering teams to illustrate key differences in collaborative versus collaboration engineering and investigates how challenges in the former can inform opportunities for the latter.

1. Introduction

Our society is built on complex systems that provide food, transportation, security, health care and modern conveniences such as entertainment, information access, and electronic communications. Creating and maintaining these systems extends far beyond the capabilities of individuals working in isolation. Collaboration has therefore become a "ubiquitous feature of organizational life." [35 p. 121]

Over the past 100 years, the focus on how people should work together has shifted from the rigid hierarchical structures advocated by scientific management [33][34] to the cross-functional, multi-disciplinary teams that form the backbone of most modern organizations [6]. The language of command and control has evolved into one of collaboration.

The word "collaboration" can be generally defined as "working together." The "work" itself can refer to many things such as physical work (e.g., raising a barn in a farming community, a sports team, a military or fire-fighting squad), social work (e.g., fund raising, policy making), the arts (e.g., theatre, orchestras) or scientific/technical efforts (construction, exploration, product development).

The concept of "together" addresses both time and place, but always implies interaction. The timescales for interaction have contracted significantly because today's communications technologies give workers virtually instantaneous access to others – and the knowledge they have produced.

The colloquial definition of collaboration, however, does not fully capture the needs of modern engineering ventures that must produce extraordinary results, with limited resources, under extreme time pressure. This paper therefore looks at two distinct but complementary areas of collaboration research: *collaborative* engineering and *collaboration* engineering. It contrasts the practice of collaborative engineering with the enabling capabilities provided by collaboration engineering to suggest how the problems encountered when collaborating on engineering projects represent opportunities for the continued evolution of collaboration engineering.

2. Background

2.1 Collaborative Engineering

Collaborative engineering is defined as "a human-centered activity that involves a dynamic socio-technical decision-making process to maximize the synergy between technical *task-work* by individuals and social *teamwork* by a collective" [24, p. 43]. This definition distinguishes between teamwork, which consists of general social processes that enable a group of people to function as a team, and task work, which consists of the discipline-specific processes necessary to achieve the goal of the team. For engineering teams, task work includes design, simulation, integration & test, interface definition, requirements definition, product decomposition, and analysis. Teamwork is independent of the type of team and includes planning, coordination, resource allocation, scheduling, and conflict resolution [32].

Consistent with the "3C" model of collaboration [12], collaborative engineering distinguishes collaboration from communication and coordination processes. Communication is an exchange of information and meaning among individuals – one-to-one, one-to-many, or in other combinations. It may occur synchronously or asynchronously via a variety of media and technology and is subject to errors, noise, and other forms of distortion that impact its effectiveness. Communication is the foundation for teamwork, and has a significant impact on team operations and performance [15].

Coordination refers to a set of processes that enable people to organize the different elements of a complex activity so they can work together effectively. Coordination includes, for example, the logistics of setting up meetings, distributing information, planning work, monitoring the performance of work, and allocating resources. Coordination processes enable individuals to function as a team and have been shown to affect team and organizational performance [28].

Coordination enables collaboration – but is not sufficient to produce collaboration. For example, in the factories of the early 20th century operating under the principles of scientific management, work was broken down into small chunks that could be performed by an individual. The efforts of all the individuals – what they worked on, when they worked on it, how they got their raw materials, and what they did with their completed product – were coordinated by supervisors and higher levels of management. While the work of individuals was coordinated, the individual workers did not "collaborate," and in fact were strongly discouraged from doing so [34].

Collaboration requires substantive interaction and joint production of results. While coordination processes may place co-workers in the same place, at the same time, with the requisite resources, collaboration occurs when these workers use their communication skills to conjointly produce results. Collaboration leverages the different perspectives, knowledge, skills, and experiences of the team members to enhance creativity, facilitate innovation, reduce costs and identify problems earlier [9]. In collaboration, team members are searching for win-win solutions that meet the combined needs of stakeholders better than individually derived or compromise solutions.

Collaborative engineering is situated within a work context, such as product development. The teams that perform product development often include members from multiple technical disciplines, marketing, and manufacturing [11], and with the advent of social media, may also include customers or other stakeholders. While the diversity of backgrounds and disciplines contribute to the advantages associated with collaboration, they are also are associated with increased levels of conflict, communication overhead, and coordination effort [16][30]. The benefits of collaboration come with a cost, which for many engineering teams is too high.

Engineering teams are not alone in facing collaboration challenges. The engineering discipline, however, often creates or uses new technologies, has projects that can be very large in scope, and require extensive knowledge across a large number of disciplines. As such, these types of teams present specific challenges for collaboration engineering as will be discussed later in this paper.

2.2 Collaboration Engineering

Collaboration engineering is defined as "an approach to designing collaborative work practices for high value recurring tasks, and transferring those designs to practitioners to execute for themselves without ongoing intervention from professional facilitators" [4, p. 119]. Collaboration Engineering is based on the fundamental premise of design patterns as applied to collaboration, i.e., recurring patterns of collaboration problems exist, and it is possible to describe reusable core solutions to these problems [35]. Six such collaboration patterns have been identified – generate, reduce, clarify, organize, evaluate, and build commitment – with the majority of attention focused on brainstorming (a generate pattern) [35].

Research on collaboration engineering consists of three main areas: the design and encoding of collaboration patterns, the role of facilitators, and the use of tools. Over the past 20 years, each of these areas has advanced in concert with the others due to the interactions between process, tools, and roles.

The purpose of collaboration engineering is to achieve similar, predictable results from groups [3], where "results" refers to group dynamics rather than the product that the group is producing. For instance, if a collaboration engineer defines a set of actions to be taken by a group to solve quality problems on a factory floor, perform a risk analysis, or come up with new ideas for a product, then the collaboration engineering is successful if using that set of actions results in similar behavior across different groups.

One mechanism for capturing *collaboration patterns* is through ThinkLets, which are facilitation techniques for predictably and repeatedly invoking known effects among people working together toward a goal [5] [35]. ThinkLets represent relatively small units of interaction, and to accomplish the goals of the group may require the sequencing of multiple ThinkLets [20]. For example, if the purpose of the group meeting is to come up with ideas for new products, the overall collaboration design may include segments for generating ideas, grouping ideas, and selecting the best for further evaluation.

Facilitators leverage collaboration patterns to support teamwork. They are experts in managing relationships, tasks and technology in a way that minimizes the cognitive load for participants while helping them attain their goals [5]. Because a facilitator guides the teamwork, the practitioners – task specialists that execute specific collaborative tasks such as risk assessment or requirements definition [20] – are able to focus on task work.

Use of a trained facilitator, separate from the team members, has been shown to be valuable [10]. Further, expert facilitators are better able than novices to adapt their collaboration process designs to accommodate surprises such as unexpected outcomes or conflict [21]. Team leaders, however, are not separate from the team, nor are they generally experts in planning facilitation processes. Research has shown a lack of interest in collaboration planning by team leaders. A comparative study of professional versus self-facilitated project teams found that team leaders did not conduct extensive planning about what tools and processes to use, and suggest this may be due to an unwillingness to spend a significant time planning the collaboration [10].

Specialized tools as well as facilitators support many collaboration processes. General purpose information and communication technologies (e.g., teleconferencing, chat, wikis, GoogleDocs) support collaboration. There are also Group Support Systems (GSS) which are tools created specifically to support collaboration (e.g., brainstorming, team writing, polling for consensus, evaluation of alternatives) [5]. GSS are designed to focus the deliberation and enhance the communication of teams working under high cognitive loads [3].

GSS technology improves meetings by making procedural structures available to groups and facilitating their use [1] and facilitating a "participative process" where all team members have the opportunity to effectively contribute [10]. Commercial groupware products now support millions of collaborators per year [2].

Most GSS research has examined special event meetings [10]. The perceived usefulness of GSS has been tightly coupled to effective facilitation, where the help of a facilitator results in high levels of perceived user friendliness and the absence of such help leads to confusion and difficulties as team members attempt to operate the tool themselves [5].

Dennis & Garfield [10] moved beyond special purpose meetings and studied interactions over time for project teams. The teams evaluated the perceived usefulness of GSS features after 7-weeks of use. The results showed that meeting memory, the facilitator, and parallelism were important while meeting planning and anonymity features were less so. The positive response to the facilitator was surprising because the team leader ran the meetings and did not ask the facilitator to do "facilitation." Instead, the facilitator provided help with the tools and answered technical questions, but otherwise simply observed the meetings. The researchers concluded (1) that project team GSS need to enable operation of the GSS with minimal training, and (2) that prior research on GSS which has been based on feature-rich software systems operated by highly trained facilitators in the context of special purpose meetings may not be appropriate for project teams and self-facilitated environments.

Collaboration engineering combines repeatable processes, tools, and facilitation to enable team collaboration. It has been highly successful in domains involving special-event type meetings, trained professional facilitators, and powerful tools, with a focus on specific collaboration patterns. In moving toward the goal of developing repeatable collaborative processes that are conducted *by the practitioners themselves* [5], collaboration engineering therefore faces three significant challenges: motivating leaders to plan their collaborations; enabling simultaneous self-facilitation and task participation, and creating tools that can be used without requiring extensive training.

2.3 Comparison

Collaboration and collaborative engineering both work to improve collaboration processes, but approach this shared goal from different perspectives, as summarized in Table 1.

Collaborative engineering is an example of an emergent knowledge process, which by definition exhibits three characteristics: "deliberations" with no best structure or sequence; highly unpredictable potential users and work contexts; and information requirements that include general, specific, *and* tacit knowledge distributed across experts and non-experts" [25, p.180]. In emergent processes, problem interpretations, deliberations, and actions unfold unpredictably. Markus et al. [25, p.182], specifically chose the term "emergent" rather than "unstructured" because the latter suggests that structuring is "possible and desirable."

The emergent character of collaborative engineering, with its inability or undesirability of structuring, at first appears diametrically opposed to collaboration engineering which seeks to exploit patterns and structure in collaboration processes. The difference, however, may be one of scale. While the overall process is not amenable to structuring, there are shorter-range planning horizons where structured

Area	Collaborative Engineering	Collaboration Engineering
Definition	A human-centered activity that involves a dynamic socio-technical decision-making process to maximize the synergy between technical task-work by individuals and social teamwork by a collective [24, p. 43]	An approach to designing collaborative work practices for high value recurring tasks, and transferring those designs to practitioners to execute for themselves without ongoing intervention from professional facilitators [4, p.119].
Approach	Improving social aspects of *engineering* task work (e.g., conceptualization, requirements definition, design, problem solving, integration & test, operations)	Facilitating collaboration processes, regardless of task domain (e.g., brainstorming, idea evaluation, structured decision making)
Goal	Help team produce a better product; win-win solutions	Help group have more-productive interactions; consensus
Team Product	Taskwork products of the team (e.g., designs, systems, consumer products, services) to satisfy a market or stakeholder need	Reusable collaboration sequences (e.g., ThinkLets), tools, and facilitation techniques, to be used by teams to for collaboration processes or for recurring work tasks
Facilita-tion	Embedded facilitation of team work by a designated or emergent leader of team; or by peers (self-managed team)	Explicit facilitation of teamwork by a professional facilitator or trained practitioner facilitator
Nature	Ad hoc, emergent work processes	Pre-planned, structured (but flexible) work processes

Table 1. Comparison of Collaboration Engineering and Collaborative Engineering

collaborations could be possible. The challenge is how to provide agile, dynamic support despite the complexity and equivocality of the work processes [18].

In collaborative engineering, collaboration can spontaneously occur at any time, in any place during day-to-day interactions between co-workers, but also punctuated by routine events such as meetings and special events such as major design reviews. This contrasts starkly with the collaboration engineering focus on special events, which bring together groups (sometimes very large groups) for hours or days for specific purposes [10]. While the application of collaboration engineering is expanding to address recurring high value tasks, such as risk analyses, the application still requires significant planning and coordination associated with an "event" rather than business as usual [10].

Collaboration engineering assigns the responsibility for planning and conducting interactions to a facilitator who is expected to have expertise in group processes as well as the use of applicable collaboration tools [20]. Such facilitators guide interactions without participating in the domain-specific task work. Because professional facilitators are not always available, there is recognized need in collaboration engineering to enable practitioners to "self-facilitate" their own collaboration processes [4]. Self-facilitation requires that a member of the team focus on teamwork (perhaps at the expense of participating in the task work), plan the collaboration, and develop skills in using collaboration tools.

Because many of the costs associated with collaboration can be attributed to poor teamwork (e.g., affective conflict, inefficiencies due to poor coordination, miscommunication [16][30]), improving collaboration via self-facilitation offers potential benefits. However, questions remain as to what type of facilitation would be most helpful in ad hoc collaborations, how to assign facilitation duties in a way that mitigates the impact of shifting that team member's attention from task to team work, and how to deploy tools in ways that minimize negative disruptions to the flow of work processes [7].

Collaborative engineering encompasses a much broader range of team interactions than currently addressed by collaboration engineering. The question addressed in the remainder of this paper is *how can the experiences gained in collaborative engineering inform collaboration engineering and vice versa?*

3. Method & Results

This research used a qualitative method, which is appropriate when performing exploratory research, and well suited to understanding the process by which events and actions take place [26]. This work was conducted following the principles of participant observation which makes it possible to "describe what goes on, who or what is involved, when and where things happen, how they occur, and why…from the standpoint of participants" [17, p.12].

The research was inspired by observations of phenomena related to collaboration made by the author during the regular execution of work duties (at a US national research laboratory as a practicing engineer, and at a university teaching graduate courses in engineering). The author's involvement can more accurately be characterized as being an "observing participant." Following completion of work duties, the author used archival records as data to support this study. Although the teams studied represent a sample convenient to the author, the cases are revelatory because the degree of access, particularly to the work team, provided "an opportunity to observe and analyze a phenomenon previously inaccessible to scientific investigation" [36, p. 40].

The teams chosen for this were the rover drivers for the Mars Exploration Rovers and student teams in a graduate engineering class. These teams represent two extreme cases which can render a phenomenon more "acutely visible" [36]. Further, these teams reside at opposite extremes relative to factors important to team performance [6]: lifespan (years vs. weeks), task difficulty (space exploration vs. school projects); product characteristics (rare science data valuable to an international science community vs. a percentage of a classroom grade), team composition (purposefully created and modified based on scope and technical needs of the project vs. assigned from pool of students in class), organizational context (complex, multi-organizational environment with evolving membership vs. single team operating within the context of a single course at a single university) and expectations for future collaboration (extending indefinitely over the course of a career vs. over the course of a semester).

The data sources for this study consist of the following, (1) for the Rover Drivers: a 2.5 hour videotaped panel interview with four original rover drivers, technical papers, public presentations, email and face-to-face discussion; (2) for the student teams: class reflection assignments, archived video of class lectures including in-class discussion, class chat, email, and project reports.

The data sources were used to create narratives describing how the two types of teams worked [cf. 13]. These narratives were then compared and contrasted to identify common themes and significant differences. The following sections present these narratives, condensed to fit within the size constraints of the paper [27].

3.1 The Mars Exploration Rover Drivers

During a panel discussion, four members of the team that drove the Mars Exploration Rovers answered student questions about their experiences collaborating. This team, in slightly modified configurations, had worked together for over 6 years to develop the visualization and planning systems that they then used to drive the rovers.

The team described how they were co-located in an open laboratory space with each person at their individual workstations. Several members of the team had worked together previously on the Mars Pathfinder mission, so brought valuable hands-on experience using tools that they had developed to do a similar mission. Other members of the team were recruited for their technical specialties, which were relevant to the enhanced features required in the next generation. For the first couple years, the team was primarily a software development team. Members described how they broke the job into constituent components, negotiated interfaces, and developed their individual pieces.

The team was sparse in that there was little excess capacity. Everyone on the team had a specific set of responsibilities that were unique to them and necessary for the success of the overall rover-driving system. Team members felt that co-location was critical to the team's success. Although they had decomposed the work into individual tasks, team members routinely needed information from others and could simply call across the room and get an immediate answer. If, for example, some aspect of a defined interface needed clarification or changing, ad hoc "meetings" involving all relevant parties were held to clear up discrepancies, assess the impacts of potential changes and propagate them through the system, and redesign or renegotiate the interfaces as needed.

The team communicated person-to-person in their work space, but also relied heavily on email, version control software, wikis and document repositories, teleconferencing, cell phones and text messaging. Different members of the team were expert on different sets of tools (e.g., version control, image processing) and facilitated their use by other members of the team.

As it got closer to the launch date for the Mars mission, the team shifted focus from software development to operations. New members joined from other parts of the project, bringing with them expertise in important areas such as the mechanical systems on the rover, the design of the robot arm, and interpreting sensor data. The expanded team of rover drivers cross-trained (mobility and arm operations were the two primary areas) and tested operations systems and processes as the spacecraft travelled to Mars.

In the time leading up to landing on Mars, the rover drivers, who had the engineering responsibility for safely operating the rovers, began interacting with the science team, which was responsible for directing the rovers to conduct scientific experiments. The rover drivers described the natural tension between these two groups as the drivers worked to keep Spirit and Opportunity safe while operating under significant environmental and engineering constraints (e.g., extremely low temperatures at night on Mars and the reliance on solar power limited how far the rover could drive on a given day without dangerously depleting its battery power), and the scientists pushed the operational limits to obtain precious science data (e.g., wanting to enter a steep-walled crater to investigate scientifically interesting features).

The drivers described the mutual learning processes that occurred over the initial 90 days on Mars as they and the scientists came to understand each other's concerns and wishes. Additional tools enabled teams to collaborate across shifts (e.g., the rover drivers used a wiki to document and share unusual events, idiosyncrasies of the twin rovers, and clever solutions). Operations processes consisted of several formal meetings each day to assess the prior day's results, determine the health, status and end location of the rover, and decide on the immediate goals for the current day's operation. They provided the mechanisms to coordinate the multiple parts of this complex mission, enabled different project members to gain insight into other functions, and kept the overall project as well as the individual teams focused on their jobs.

Following completion of the primary 90 day mission, during which the rovers met all science goals established by NASA, the nature of operations shifted. Rather than working around the clock "on Mars time" – processes were changed to work on a routine day-shift schedule. Many of the scientists who had collocated to the operations center returned to their home institutions. Members of the development team either dropped to part time or moved on completely to the next project.

New members were selected and trained to staff the rover driver positions. These new team members went through a rigorous training and apprentice process before being allowed to "drive" the rovers. The original rover drivers learned their job on Earth in testbeds and simulated environments, but then honed their expertise via on-the-job driving on Mars and by dealing with the multitude of unexpected conditions, idiosyncrasies in the performance of equipment, challenging new ways the scientists wanted to use the rovers, and ever-changing environmental conditions. New team members had to learn the basics, as well as integrate the enormous body of knowledge gained while operating on Mars.

While the technical challenges were significant, these new recruits also faced the social challenges of understanding the different people involved in the project, what their roles were, how these roles evolved, and the history that made individuals more or less sensitive to specific issues. The original team worked with these people under battlefield type conditions during the exhilarating early months of the mission, building strong relationships and a body of shared experience. The new recruits, however, were faced with disembodied voices on speaker phones that were continually being interrupted by the pressing demands of their other jobs. Periodic visits by distributed team members helped form personal connections, but these were not the rich, intense bonds that helped the original team to deal with conflicts and misunderstandings. The rover drivers described how, to humanize the process, they pasted pictures of the different scientists onto popsicle sticks and stood them up around the table during teleconferences to represent the remote attendees.

Throughout the continuing mission, project managers moved in and out of the project, and the rover drivers had a similar transition in leadership. They describe the hand-off of leadership as being straightforward, and the team continues its work as Opportunity roves across Mars as the longest operating robot on the surface of another planet.

3.2 Student Teams in a Graduate Engineering Class

Nineteen masters and Ph. D. students in a graduate course in collaborative engineering participated in a series of four team projects over the course of a semester. The class consisted of students taking the course on campus as well as others taking the course remotely. Students spanned multiple time zones, two engineering departments, were split between full time/part time with jobs and US/international students, and varied in demographic characteristics.

The class projects were designed to give the students a variety of shared experiences relevant to material covered in the course. Teams of 4-5 students were assigned for the first two projects by the professor to create teams that had a similar mix of the characteristics listed above. For the Project 1, teams were tasked with identifying customer needs for a futuristic product (e.g., bus stop kiosk, home security system). Project 2 continued the product development cycle to develop functional requirements and produce a paper prototype. Projects 3 and 4 were completely separate efforts to produce reports; students were allowed to self-select their teams and topics, but encouraged to have at least 3 members. For all projects, students received a team grade based on the product they developed and an individual grade for a reflection assignment analyzing their experiences working on their team.

Although collaboration tools were available to the teams through the University's Blackboard system, the teams chose publicly available tools such as Skype (audio conferencing), personal email, Google Docs (co-authoring), and Doodle (scheduling) to support their work. In addition, some teams took advantage of corporate teleconferencing facilities through their working members. Team members reported different levels of competence in using tools, and many students described how a more-experienced team member tutored them in how to use the tool, including step-by-step instructions for setting up an account, configuring the tool for their use, and trouble-shooting when problems occurred. Students reported general satisfaction with the tools they used despite experiencing some problems with the individual technologies.

While the tools were sufficient for their projects, students expressed varying degrees of frustration with their teams' basic coordination processes. Teams routinely had difficulties finding a mutually acceptable meeting time, members were late or missed meetings, task assignments were duplicated or missed completely, team members didn't deliver their work on time, individuals would edit the wrong versions of documents, and team members interpreted project instructions inconsistently.

Team start-up experiences varied significantly. Some team members reported that a team member took charge, coordinated their activities, and helped the team quickly reach decisions on work processes and assignments. Other team members reported that their team wasted a lot of time socializing and that there were power struggles for who should lead the team. In several cases, opposing views were given by different students on the same team.

Several teams struggled with not having an assigned leader. Students reported conflict when multiple team members vied for leadership, and inefficiency when no one stepped into the leadership role. Students distinguished between leadership with respect to organizing the work, preparing minutes, and integrating team products (which was considered positive) and intellectual leadership where the "leader" made design decisions, dictated a specific course of action, determined what ideas were "heard" or treated questions as attacks and prevented discussion (which were considered negative).

Project 2, by design, changed team membership while maintaining continuity of the project. Conflicts occurred on one team because of different interpretations of the starting conditions: the single team member continuing from Project 1 considered new members to be joining his Project 1 team; conversely, two team members that had previously worked together on a different project viewed themselves as the nucleus of the team but with a new task. In their individual reflection assignments, one team member described the frustration with team members for wanting to redo prior work, while others took credit for their patience in allowing that team member the opportunity to explain the rationale behind choices made in the prior project.

Although the projects were structured to encourage collaboration, most student teams quickly worked to divide the assignment into individual tasks. For those teams, meetings focused on coordination, rather than collaboration, and the majority of students reported significant conflicts, stress, and the need for heroic individual efforts to integrate the final product. Some team members reported that they felt they were not listened to, and that their ideas and concerns were dismissed.

One team, however, reported in their individual reflections and in class discussion that they intentionally approached the project as a collaborative effort. They described how leadership shifted between members as a person's individual expertise made them the best person to lead a particular part of the project. They reported having team discussions to test out ideas, make sense of project instructions, and jointly make design decisions. They worked together to easily integrate individual results into the final product. Several of the team members described this project as the best team experience they had in their life.

When students were given the opportunity to self-select for projects 3 and 4, the teams were smaller (2-3 members), and consisted of members that had prior positive experiences working together. Despite the variety in reported experiences, each of the teams produced high quality products that met all requirements of the assignment.

4. Discussion

4.1 Leadership and Facilitation Roles

The original lead rover driver was selected by management to fill that position. He had performed the same function on the previous Mars mission and had extensive, unique experience relevant to the task. He also recruited his team members specifically to take advantage of their individual areas of expertise. When he stepped down, his replacement was widely regarded as the obvious choice due to his contributions to the development of the tools used to drive the rover and his multiple years of experience gained on the project. In both cases, legitimacy of leadership was official (from the organization) and social (based on the character of the individual).

In contrast, the student teams were leaderless. Some students perceived this as a void that needed to be filled. Others took advantage of emergent characteristics of the project to choose the most appropriate person at that time. No student had specialized skills or knowledge that would make him or her an obvious choice. For the majority of teams, the legitimacy of the leader was neither official nor social, and students seemed to step into that role based on their personal intolerance for a lack of leadership.

When team members reported positive feelings about leadership, it was for behaviors that were more characteristic of a facilitator rather than a leader: assistance with tools, helping to coordinate activities, integrating products, and keeping team members informed. These experiences are similar to those reported by [10] where tool support was credited with leading to the perceived usefulness of the facilitator. Building on the work of [29], they suggest that skilled users of tools can be thought of as "appropriation agents" who shape the way in which teams appropriate a tool for their use.

Hackman [14] emphasizes the importance of *leadership* rather than a *team leader*. In an extensive study of intelligence teams, he found that the best team leaders practice shared leadership by actively encouraging contributions from the members of the team. Further, he found that one aspect of shared leadership, *peer coaching*, defined as "the degree to which team members taught, helped and learned from one another" (p. 165), was significantly associated with team effectiveness.

4.2 Task Decomposition and Integration

Both rover drivers and student teams demonstrated a proclivity to quickly decompose the *project* into smaller tasks that could be performed by individuals – and decompose the *product* into constituent parts for future integration based on the tasks. There are always multiple options for how to decompose a task, and to the extent that there is a "best" option, it will be based not just on the task, but on other factors such as the composition of the team and the culture of the organization. Given that teams are predisposed to start work by decomposing the task, then there are two potential goals for this important process:
(1) complete the decomposition and assignment as efficiently as possible, so as to quickly identify those remaining parts of the task that require collaboration,
(2) identify those parts of the work that could be performed independently, but there are benefits to performing it collaboratively.

Simply put, teams need to identify tasks where they *must* collaborate and those where they *should* collaborate, i.e., they may benefit by also decomposing their collaboration process [cf. 31]. By identifying these sooner, teams can more adequately allocate their resources and avoid major problems that occur when the work must be integrated.

4.3 Membership Transitions

The rover driver team changed significantly over the course of its nearly decade-long existence. While incidental changes were accommodated informally, a major change occurred when a new cadre of replacement drivers was recruited after completion of the primary Mars mission. The existing rover drivers put in place a program of on the job training that involved learning by shadowing qualified team members, followed by a period of supervised operations before a driver was allowed to drive solo (this process could take as long as one year).

The rover drivers evolved a process for helping new team members learn the knowledge constructed by the original rover drivers during the entire course of the project. The training process consisted primarily of experiential learning [19] with a large dose of team socialization [22] to learn not only the technical details, configuration and state of the rovers on Mars, but also the tools and processes used to drive the rovers and the complex network of people involved in the overall mission.

Cramton [8], building on [23], would classify this large volume of information as mutual knowledge – the knowledge that communicating parties share in common and know they share. She proposes that maintaining mutual knowledge is a central problem for geographically dispersed collaboration and that five types of failures of mutual knowledge can occur: failure to communicate and retain contextual information, unevenly distributed information, difficulty communicating and understanding the salience of information, differences in speed of access to information, and difficulty interpreting the meaning of science. While the original rover drivers built up this knowledge over time through direct shared experiences with others on the project, the new drivers had to learn at a significantly accelerated pace and experienced the problems associated with failures of mutual knowledge.

On a much smaller scale, the student teams also experienced mutual knowledge failures due to geographic dispersion. These failures, similar to those described by [8] in a study of student teams, led to similar negative attitudes toward team members, attributing problems personally rather than situationally in ways that were non-constructive for continued communication, and team members not wanting to work with each other in the future.

4.4 Implications for Collaboration Engineering

Each of the areas discussed above offer opportunities for the extension of collaboration engineering. The recurrence of problems across two widely differing classes of collaborative engineering teams suggests that these are common problems that may be amenable to a common solution approach.

Based on experiences with facilitation, collaboration engineering could offer important insights in the realization of shared leadership. In addition, given that team members routinely facilitate the adoption and use of tools, lessons learned from GSS research could enhance member performance of these roles.

Task decomposition and membership transitions represent two classes of recurring collaborative processes that move beyond the "generate" pattern. Collaboration engineering offers the ability to exploit the patterns in this work and provide ways to accomplish these tasks more efficiently and with greater levels of participation from team members. Addressing these areas would significantly challenge the fundamental principles of collaboration engineering because these activities take place over long periods of time, without the full participation of the team in a special event.

5. Conclusion

This paper contributes to the collaborative and collaboration engineering fields in three ways. First, it distinguishes between these two related but distinct disciplines. Because of the similarity in names, people may not realize that they represent different approaches to collaboration. By calling attention to their significant differences, this work begins to build a bridge between the two disciplines.

Second, this work provides insight into other types of collaboration (as called for in [35]). Because so much collaboration happens in emergent and ad hoc ways, these processes often remain hidden from researchers using less immersive techniques [17]. Finally, this paper offers rich examples, grounded in real-world team experiences that suggest where collaboration engineering could inform collaborative engineering and vice versa.

This work is clearly in its infancy, and therefore is limited in terms of its generalizability and applicability to other domains. The examples are gleaned from the observations of the author during the execution of her organizational duties and are therefore limited to the areas where she worked and the job-driven techniques used to record observations.

Despite these limitations, it is hoped that this work can open a dialogue between the two communities and lead to more efficient and effective collaboration.

6. Acknowledgements

This research was carried out in part at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

7. References

[1] Bostrom, R.P.; Anson, R.; and Clawson V.K. Group facilitation and group support systems. In L.M. Jessup and J.S. Valacich (eds.), *Group Support Systems: New Perspectives* . New York: Macmillan, 1993, 146-168.

[2] Briggs, R.O., On theory-driven design and deployment of collaboration systems. *International Journal of Human-Computer Studies, 64* (7), July 2006, 573-582

[3] Briggs, R.O., Nunamaker, J. & Sprague, R. 1001 Unanswered Research Questions in GSS. *Journal of Management Information Systems, 14*(3), 1998, 3-22.

[4] Briggs, R.O., Vreede, G-J de, & Massey, A.P. Introduction to the JAIS special issue on collaboration engineering, *Journal of the Association for Information Systems, 10* (Special Issue), March 2009, 118-120.

[5] Briggs, R.O., Vreede, G-J de, & Nunamaker, J.F. Collaboration Engineering with ThinkLets to pursue sustained success with group support systems, *Journal of Management Information Systems, 19* (4), 2003, 31-63.

[6] Cohen, S. G. & Bailey, D. E. "What makes teams work: Group effectiveness research from the shop floor to the executive suite." *Journal of Management, 23*(3), 1997, 239-290.

[7] Cooper, L. P. A research agenda to reduce risk in new product development through knowledge management: A practitioner perspective. *Journal of Engineering and Technology Management, 20,* 2003, pp. 117-140.

[8] Cramton, C.D., The mutual knowledge problem and its consequences for dispersed collaboration. Organization Science, 12(3), May-June 2001, 346-371.

[9]Dailey, R. C., The role of team and task characteristics in R&D team collaborative problem solving and productivity. Management Science, 24(15), Nov. 1978, 1579-1588.

[10] Dennis, A.R. & Garfield, M. J. The adoption and use of GSS in project teams: Toward more participative processes and outcomes. *MIS Quarterly, 27*(2), 2003, 289-323.

[11] Dougherty D. (1992). Interpretative barriers to successful product innovation in large firms. *Organization Science, 3*(2), 179-202.

[12] Fuks, H., Raposo, A.B., Gerosa, M.A., & Lucena, C.J.P. "Applying the 3C model to groupware development." *International Journal of Cooperative Information Systems 14*(2-3), 2005, 299-328.

[13] Gersick C. Time and transition in work teams: Toward a new model of group development. *Academy of Management Journal, 31*, (1988). 9-41.

[14] Hackman, J. R. *Collaborative Intelligence: Using Teams to Solve Hard Problems*. 2011, San Francisco: Berrett-Koehler Publishers, Inc.

[15] Hirokawa R. Y., A comparative analysis of communication patterns within effective an[d] ineffective decision-making groups. *Communication Monographs, 47*, 1980, 312-321.

[16] Jehn, K. A. & Mannix, E. A. The dynamic nature of conflict: A longitudinal study of intragroup conflict and group performance. *Academy of Management Journal. 44*(2), 2001, 238-251.

[17] Jorgensen, D. L. *Participant Observation: A Methodology for Human Studies*. Newbury Park, CA: Sage; 1989. (Applied Social Research Methods Series; 15).

[18] Helquist, J.H., Deokar, A., Meservy, T. & Kruse, J. "Dynamic collaboration: Participant-driven agile processes for complex tasks." *The DATA BASE for Advances in Information Systems, 42*(2), 2011, 95-115.

[19] Kolb, D. A. *Experiential Learning: Experience as the Source of Learning and Development.* Englewood Cliffs, NJ: Prentice Hall, 1984.

[20] Kolfschoten, G. L., Briggs, R.O., Vreede, G-J de, Jacobs, P.H.M, and Appelman, J. H. A conceptual foundation of the thinkLet concept for Collaboration Engineering, *International Journal of Human-Computer Studies, 16*(4), 2007, 347-361.

[21] Kolfschoten, G. L., Hengst-Bruggeling, M. den, & Vreede, G-J de Issues in the design of facilitated collaboration processes, *Group Decision and Negotiation, 64*(7), July 2006, 611-621.

[22] Kozlowski, S. W. J., Watola, D.J., Jensen, J. M., Kim, B.H, & Botero, I. C. Developing adaptive teams: A theory of dynamic team leadership. In E. Salas, G. F. Goodwin and C. S. Burke (Eds), *Team Effectiveness in Complex Organizations: Cross-Disciplinary Perspectives and Approaches*, 2009, New York: Psychology Press.

[23] Krauss, R. P, & Fussell, S. Mutual knowledge and communicative effectiveness. In Galegher, J., Kraut, R., & Egido, C. (Eds), *Intellectual Teamwork: Social and Tehcnological Foundations of Cooperative Work.* Hillsdale, NJ: Lawrence Erlbaum, 111-146.

[24] Lu, S. C-Y. "Collective rationality of group decisions in collaborative engineering." *International Journal of Collaborative Engineering, 1*(1/2), 2009, 38-74.

[25] Markus, M.L., Majchrzak, A. & Gasser, L., "A design theory for systems that support emergent knowledge processes." *MIS Quarterly. 26*(3), 2002, 179-212

[26] Maxwell, J. A. *Qualitative Research Design: An Interactive Approach*. Thousand Oaks, CA: Sage; 1996. (Applied Social Research Methods Series; 41).

[27] Myers, M. D. "Investigating information systems with ethnographic research." *Communications of the Association for Information Systems, 2*, Article 23, 1999.

[28] Olson, G. M., Malone, T. W., & Smith, J. B. *Coordination Theory and Collaboration Technology*. 2001. Psychology Press.

[29] Orlikowski, W. "Using technology and constituting structures: A practice lens for studying technology in organizations, *Organization Science, 11*(4), 2000, 404-428.

[30] Pelled, L. H., & Adler, P. S. Antecedents of Intergroup Conflict in Multifunctional Product Development Teams - a Conceptual-Model. *IEEE Transactions on Engineering Management, 41*(1), 1994, 21-28.

[31] Santanen, E., Kolfschoten, G., & Golla, K. The Collaboration Engineering Maturity Model. In *Proceedings of Hawaii International Conference on System Sciences – 39*. Kauai, HI, 2006.

[32] Stevens, M. J., & Campion, M. A. The knowledge, skill, and ability requirements for teamwork: Implications for human resource management. *Journal of Management, 20*(2), 1994, 503-530.

[33] Taylor, F., *Principles of Scientific Management*. 1911, New York: Harper & Brothers.

[34] Turner, W.C., Mize, J.H., Case, K.E., & Nazemetz, J. W. *Introduction to Industrial and Systems Engineering, 3rd Edition.* 1993, Englewood Cliffs, NJ: Prentice-Hall.

[35] Vreede, G-J de, Briggs, R.O. & Massey, A.P. Collaboration engineering: Foundations and opportunities: Editorial to the Special Issue, *Journal of the Association for Information Systems, 10* (Special Issue), March 2009, 121-137.

[36] Yin, R.K. *Case Study Research, Design and Methods, 2nd Edition*. 1994, Newbury Park: Sage Publications.